

PATENT

IN THE CLAIMS:

Please amend claims 1-3, 6-8, 11-12, 18-20, 23, and 24 as indicated in the following:

1. (Currently Amended) A method comprising:
 - receiving encrypted data from a [[first]] plurality of applications including a first encrypted data from a first application assigned to a first key register and a second encrypted data from a second application assigned to a second key register;
 - assigning a third key register for decrypting data from the first application based upon a request for re-authentication;
 - receiving a third encrypted data from the first application assigned to a third key register;
 - and
 - providing the first encrypted data to a first decryption source after receiving the third encrypted data.
2. (Currently Amended) The method as in Claim 1, further including:
 - providing the third encrypted data to the first decryption source; and
 - providing the second encrypted data to [[the]]a second decryption source.
3. (Currently Amended) The method as in Claim 1, wherein the first decryption source decrypts the first encrypted data using a first encryption key stored in the first key register.
4. (Original) The method as in Claim 1, wherein the request for re-authentication is a notification sent by the first application to a driver.
5. (Previously Presented) The method as in Claim 1, wherein assigning the third key register includes locating an unused key register.

PATENT

6. (Currently Amended) A method of providing multiple channels of secure multimedia data, the method comprising:

- receiving a first authentication request from a first multimedia application;
- receiving a second authentication request from a second multimedia application, wherein the second multimedia application is different than the first multimedia application;
- assigning a first key register to the first multimedia application based upon the first authentication request;
- assigning a second key register to the second multimedia application based upon the second authentication request;
- receiving first encrypted data based upon a first encryption key from the first multimedia application; and
- receiving second encrypted data based upon a second encryption key from the second multimedia application, wherein the first and second encrypted data are for simultaneous real-time play back.

7. (Currently Amended) The method as in Claim 6, wherein the first and second applications are the same application.

8. (Currently Amended) The method as in Claim 6, wherein the first and [[the]] second applications are capable of providing a notification to the driver.

9. (Previously Presented) The method as in Claim 8, wherein the notification includes the first and second authentication requests.

10. (Original) The method as in Claim 8, wherein the notification includes a request for re-authentication.

11. (Currently Amended) The method as in Claim 6, wherein the first and [[the]] second multimedia applications relate to video applications.

PATENT

12. (Currently Amended) The method as in Claim 6, wherein assigning the first and [[the]] second encryption keys includes selecting unused key registers.
13. (Original) The method as in Claim 6, wherein the first and second key registers are stored in a driver.
14. (Original) The method as in Claim 6, wherein the first and second key registers are stored in hardware.
15. (Previously Presented) The method as in Claim 6, further including providing a binary file to developers of the first and second multimedia applications for inclusion in the first and second multimedia applications.
16. (Original) The method as in Claim 15, wherein the binary file is for decoding commands generated in the first and second multimedia applications to hardware commands.
17. (Original) The method as in Claim 15, wherein the binary file includes a set of encryption keys for encrypting data generated in the first and second applications.

PATENT

18. (Currently Amended) A system comprising:

a data processor having a first I/O buffer;

a memory having a second I/O buffer coupled to the first I/O buffer of the data processor,
the memory capable of storing code for:

a driver for:

receiving a first authentication request from a first multimedia application;

receiving a second authentication request from a second multimedia
application;

assigning a first key register to the first multimedia application based upon
the first authentication request;

assigning a second key register to the second multimedia application based
upon the second authentication request[[.]];

receiving first encrypted data based upon a first encryption key from the
first multimedia application; and

receiving second encrypted data based upon a second encryption key from
the second multimedia application, wherein the first and second
encrypted data are for simultaneous real-time play back; and

a hardware device for processing data generated by the first and second multimedia
applications including:

a key register for storing a decryption key;

a decryption component for decrypting data using said decryption key; and

a processing component for processing multimedia data.

19. (Currently Amended) The system as in Claim 18, wherein the ~~plurality of first and second~~
multimedia applications include a binary file for encrypting data generated within the ~~plurality~~
~~of first and second~~ multimedia applications.

20. (Currently Amended) The system as in Claim 19, wherein the binary file is further capable of
decoding data generated within the ~~plurality of first and second~~ multimedia applications to
generate hardware commands.

PATENT

21. (Previously Presented) The system as in Claim 18, wherein the driver is further capable of:
- decrypting the first encrypted data based on the first encryption key;
 - decrypting the second encrypted data based on the second encryption key;
 - encrypting the first and second encrypted data using a hardware key to generate a third encrypted data; and
 - providing the third encrypted data to the hardware device.
22. (Previously Presented) The system as in Claim 18, wherein the hardware device includes sets of key registers for storing a plurality of decryption keys and the hardware device is further capable of:
- decrypting the first encrypted data based on the first encryption key; and
 - decrypting the second encrypted data based on the second encryption key.
23. (Currently Amended) A computer readable medium tangibly embodying a plurality of programs of instructions, the plurality of programs including:
- a driver for:
 - receiving a first authentication request from the first multimedia application;
 - receiving a second authentication request from the second multimedia application;
 - assigning a first key register to the first multimedia application based upon the first authentication request;
 - assigning a second key register to the second multimedia application based upon the second authentication request;
 - receiving first encrypted data based upon a first encryption key from the first multimedia application; and
 - receiving second encrypted data based upon a second encryption key from the second multimedia application, wherein the first and second encrypted data are for simultaneous real-time play back.

PATENT

24. (Currently Amended) The computer readable medium as in Claim 22, wherein the plurality of programs further include a plurality of multimedia applications including ~~[[a]]~~the first multimedia application and ~~[[a]]~~the second multimedia application, wherein the second multimedia application is different from the first multimedia application.
25. (Original) The computer readable medium as in Claim 24, wherein the plurality of multimedia applications include a binary file for encrypting data generated within the plurality of multimedia applications.
26. (Original) The computer readable medium as in Claim 24, wherein the binary file is further capable of decoding data generated within the plurality of multimedia applications to generate hardware commands.
27. (Original) The computer readable medium as in Claim 24, wherein the driver is further capable of:
decrypting the first encrypted data based on the first encryption key; and
decrypting the second encrypted data based on the second encryption key.
28. (Previously Presented) A method comprising:
providing a binary file to an application vendor, wherein the binary file is for:
providing a method of negotiating encryption with a device driver;
generating an encryption key value based upon a negotiation with the device driver; and
providing an encryption of data using a final key value.